EXHIBIT 18

International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.993.1

(06/2004)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

Digital sections and digital line system – Access networks

Very high speed digital subscriber line transceivers

ITU-T Recommendation G.993.1

ITU-T G-SERIES RECOMMENDATIONS

TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TESTING EQUIPMENTS	G.500–G.599
TRANSMISSION MEDIA CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
General	G.900–G.909
Parameters for optical fibre cable systems	G.910–G.919
Digital sections at hierarchical bit rates based on a bit rate of 2048 kbit/s	G.920–G.929
Digital line transmission systems on cable at non-hierarchical bit rates	G.930–G.939
Digital line systems provided by FDM transmission bearers	G.940–G.949
Digital line systems	G.950–G.959
Digital section and digital transmission systems for customer access to ISDN	G.960–G.969
Optical fibre submarine cable systems	G.970–G.979
Optical line systems for local and access networks	G.980–G.989
Access networks	**G.990–G.999**
QUALITY OF SERVICE AND PERFORMANCE - GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DIGITAL TERMINAL EQUIPMENTS	G.7000–G.7999
DIGITAL NETWORKS	G.8000–G.8999

For further details, please refer to the list of ITU-T Recommendations.

ITU-T Recommendation G.993.1

Very high speed digital subscriber line transceivers

Summary

G.993.1 VDSL (Very high speed Digital Subscriber Line) permits the transmission of asymmetric and symmetric aggregate data rates up to tens of Mbit/s on twisted pairs. G.993.1 includes worldwide frequency plans that allow asymmetric and symmetric services in the same group of twisted pairs (known as a binder). G.993.1 transceivers must overcome many types of ingress interference from radio and other transmission techniques that occur in the same frequencies of typical deployment scenarios. Similarly, G.993.1 transmission power transmission levels have been designed to minimize potential egress interference into other transmission systems. As with other Recommendations in the G.99x series, G.993.1 uses G.994.1 to handshake and initiate the transceiver training sequence.

Source

ITU-T Recommendation G.993.1 was approved on 13 June 2004 by ITU-T Study Group 15 (2001-2004) under the ITU-T Recommendation A.8 procedure.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

CONTENTS

7.2.1 Data flow

The eoc data flow includes two contra-directional streams of 2-octet blocks each (eoc_tx, eoc_rx) with independent rates flowing between the eoc application layer (VME) and TPS-TC OC block (OC-TC). The bit rates of both streams shall not exceed the predefined upper limit of the OC channel aggregate transport capability. The data flow signal description is presented in Table 7-1.

If data streams are *serial* by implementation, the MSB of each octet shall be sent first.

7.2.2 Synchronization flow

This flow provides synchronization between the eoc application layer (VME) and the OC-TC (see 10.3.1). The flow includes the following synchronization signals, presented in Table 7-2:

- transmit and receive timing signals (eoc_tx_clk, eoc_rx_clk): both asserted by the eoc processor;
- transmit enable flag (tx_enbl): asserted by OC-TC and allows to transmit the next 2-octet block;
- receive enable flag (rx_enbl): asserted by OC-TC and indicates that the next 2-octet block is allocated in the OC-TC receive buffer.

Table 7-2/G.993.1 – OC-TC: γ interface data and synchronization flow summary

Signal	Description	Direction	Notes
Data flow			
eoc_tx	Transmit eoc data	VME → OC-TC	Two-octet block
eoc_rx	Receive eoc data	VME ← OC-TC	
Synchronization flow			
eoc_tx_clk	Transmit clock	VME → OC-TC	
eoc_rx_clk	Receive clock	VME → OC-TC	
tx_enbl	Transmit enable flag	VME ← OC-TC	
rx_enbl	Receive enable flag	VME ← OC-TC	

NOTE – The main buffering required to implement the eoc communication protocol should be provided by the VME; only a minimum buffering for eoc is supposed in OC-TC.

8 PMS-TC sublayer

The PMS-TC sublayer provides transmission medium specific TC functions, such as framing, Forward Error Correction (FEC), and interleaving.

8.1 PMS-TC functional model

All data bytes shall be transmitted MSB first. All serial processing (e.g., scrambling, CRC calculation) shall however be performed LSB first, with the outside world MSB considered as the VDSL LSB. As a result, the first incoming bit (outside world MSB) shall be the first bit processed inside VDSL (VDSL LSB). The PMS-TC functional diagram is presented in Figure 8-1.

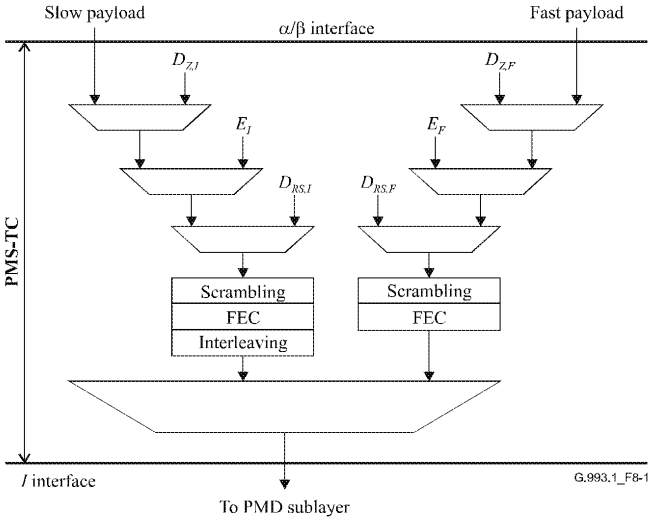

Figure 8-1/G.993.1 – Diagram of PMS-TC sublayer

8.2 Scrambler

A scrambler shall be used to reduce the likelihood that a long sequence of zeros will be transmitted over the channel. The scrambler shall be self-synchronizing such that descrambling can occur without requiring a particular alignment with the scrambled sequence. The scrambler shall be represented by the equation below, where *m(n)* is a message bit sample at sample time *n* and the output of the scrambler *x(n)* shall be given by:

$$x(n) = m(n) + x(n-18) + x(n-23)$$

All arithmetic shall be modulo 2. As long as the scrambler is initialized with values other than zero, an "all zeros" sequence for *m(n)* will result in a pseudo-random sequence of length $2^{23} - 1$.

8.3 Forward error correction

A standard byte-oriented Reed-Solomon code shall be used to provide protection against random and burst errors.

A Reed-Solomon code word contains $N = K + R$ bytes, comprised of R redundant check bytes c_0, c_1, …, c_{R-2}, c_{R-1} appended to K message bytes m_0, m_1, …, m_{K-2}, m_{K-1}. The check bytes shall be computed from the message bytes using the equation

$$C(D) = M(D)D^R \bmod G(D)$$

where:

$M(D) = m_0 D^{K-1} \oplus m_1 D^{K-2} \oplus \ldots \oplus m_{K-2} D \oplus m_{K-1}$ is the message polynomial

$C(D) = c_0 D^{R-1} \oplus c_1 D^{R-2} \oplus \ldots \oplus c_{R-2} D \oplus c_{R-1}$ is the check polynomial

$G(D) = \prod (D \oplus \alpha^i)$ is the generator polynomial of the Reed-Solomon code, where the index of the product runs from i = 0 to $R - 1$.

This means that $C(D)$ is the remainder obtained from dividing $M(D)D^R$ by $G(D)$. The arithmetic shall be performed in the Galois Field GF(256), where α is a primitive element that satisfies the

primitive binary polynomial $x^8 \oplus x^4 \oplus x^3 \oplus x^2 \oplus 1$. A data byte $(d_7, d_6, \ldots, d_1, d_0)$ is identified with the Galois Field element $d_7\alpha^7 \oplus d_6\alpha^6 \oplus \ldots \oplus d_1\alpha \oplus d_0$.

Both K and R shall be programmable parameters. Redundancy values of R = 0, 2, 4, 6, 8 … 16 shall be supported. The following codeword parameters specified as (N,K) shall be supported: (144,128) and (240,224). Other values for N and K are optional. However, N shall be less than or equal to 255.

8.4 Interleaving

8.4.1 General

Interleaving shall be used to protect the data against bursts of errors by spreading the errors over a number of Reed-Solomon codewords. The interleave depth shall be programmable with a maximum interleave depth of 64 codewords when the number of octets per codeword (N) equals 255. For smaller values of N the interleave depth can grow nearly proportionately.

It shall be possible to adjust the interleave depth via the management system to meet latency requirements. The latency of the slow path is a function of the data rate and burst error correction capability. For data rates greater than or equal to 13 Mbit/s, the latency between the α and β interfaces shall not exceed 10 ms when the interleaver depth is set to the maximum. At lower data rates there is a trade-off between higher latency and decreased burst error correction ability. At any data rate, the minimum latency occurs when the interleaver is turned off.

When the interleaver is on, the codewords shall be interleaved before transmission to increase the immunity of RS codewords to bursts of errors. The convolutional interleaver is defined by two parameters: the interleaver block length, I, and the interleaving depth, D. The block length I shall divide the RS codeword length N (i.e., N shall be an integer multiple of I). The convolutional interleaver uses a memory in which a block of I octets is written while an (interleaved) block of I octets is read. Details of the implementation are given in 8.4.2.

The same size interleaving memory (see Table 8-1) is needed for interleaving at the transmitter and de-interleaving at the receiver.

The convolutional interleaving introduces an absolute read-to-write delay, Δ_J, that increments linearly with the octet index within a block of I octets:

$$\Delta_j = (D-1) \times j$$

where j = 0, 1, 2, …, I – 1.

8.4.2 Triangular implementation

To decrease the implementation complexity, the delay increment $(D-1)$ shall be chosen as a multiple of the interleaver block length (I), i.e.: $D-1 = M \times I$. The $(D-1)$ to I ratio is the interleaving depth parameter (M). The characteristics of convolutional interleaving are shown in Table 8-1. The parameters t and q depend on the characteristics of the RS code and are defined as:

– t = number of bytes that can be corrected by RS codewords = half the number of redundancy bytes = $R/2$;

– q = length of RS codeword divided by the length of an interleaver block = N/I.

Table 8-1/G.993.1 – Characteristics of convolutional interleaving

Parameter	Value
Interleaver block length (I)	I bytes (equal to or divisor of N)
Interleaving Depth (D)	$M \times I + 1$
(De)interleaver memory size	$M \times I \times (I - 1)/2$ bytes
Correction capability	$\lfloor t/q \rfloor \times (M \times I + 1)$ bytes
End-to-end delay	$M \times I \times (I - 1)$ bytes

The example in Figure 8-2 shows $I = 7$. I parallel branches (numbered $0 \ldots I - 1$) are implemented with a delay increment of M octets per branch. Each branch shall be a FIFO shift register (delay line) with length $0 \times M \ldots (I - 1) \times M$ bytes. The deinterleaver is similar to the interleaver, but the branch indices are reversed so that the largest interleaver delay corresponds to the smallest deinterleaver delay. Deinterleaver synchronization shall be achieved by routing the first byte of an interleaved block of I bytes into branch 0.

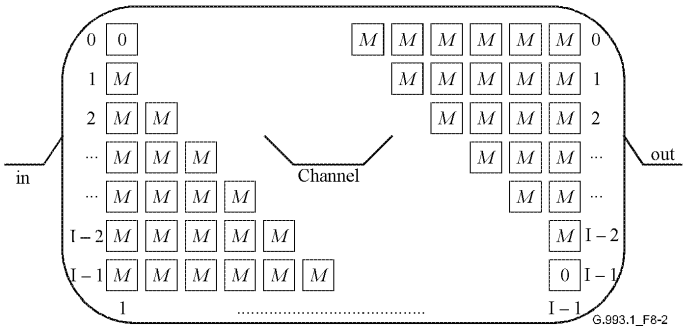

Figure 8-2/G.993.1 – Implementation example with $D - 1 = M \times I$ and $I = 7$

Table 8-2/G.993.1 – Example of interleaver parameters with RS(144,128)

Rate [kbit/s]	Interleaver parameters	Interleaver depth	(De)interleaver memory size	Erasure correction	End-to-end delay
50 × 1024	$I = 72$ $M = 13$	937 blocks of 72 bytes	33 228 bytes	3 748 bytes 520 µs	9.23 ms
24 × 1024	$I = 36$ $M = 24$	865 blocks of 36 bytes	15 120 bytes	1 730 bytes 500 µs	8.75 ms
12 × 1024	$I = 36$ $M = 12$	433 blocks of 36 bytes	7 560 bytes	866 bytes 501 µs	8.75 ms
6 × 1024	$I = 18$ $M = 24$	433 blocks of 18 bytes	3 672 bytes	433 bytes 501 µs	8.5 ms
4 × 1024	$I = 18$ $M = 16$	289 blocks of 18 bytes	2 448 bytes	289 bytes 501 µs	8.5 ms
2 × 1024	$I = 18$ $M = 8$	145 blocks of 18 bytes	1 224 bytes	145 bytes 503 µs	8.5 ms

The following interleaver parameters shall be supported:

– For $(N,K) = (144,128)$ the following values for M and I shall be supported:
  $I = 36$ and M between 2 and 52.

– For $(N,K) = (240,224)$ the following values for M and I shall be supported:
  $I = 30$ and M between 2 and 62.

8.5 Framing

8.5.1 Frame description

A *frame* is a set of bytes carried by one DMT symbol. The frame frequency depends on the total length of the cyclic extension (see 9.2.2). A frame shall be composed of two sources: the "fast" buffer and the "interleaved" (or "slow") buffer. The index i refers to parameters related to the fast or interleaved buffers ($i \in \{F, I\}$). The inclusion of the fast buffer shall be optional. When the fast buffer is not included, the interleaved buffer shall have the capability to carry non-interleaved data by setting the interleaver depth to zero.

Both fast and interleaved buffer shall contain an integer number of RS-encoded bytes. Neither the fast nor the interleaved buffer is required to carry an integer number of RS codewords. To reduce the end-to-end delay, it is recommended that the fast buffer (or the interleaved buffer when the interleaver depth is zero) carries at least one RS codeword. The framing parameters shall be exchanged between the VTU-O and VTU-R during initialization.

The framing rules described in this clause are represented in Figure 8-3.

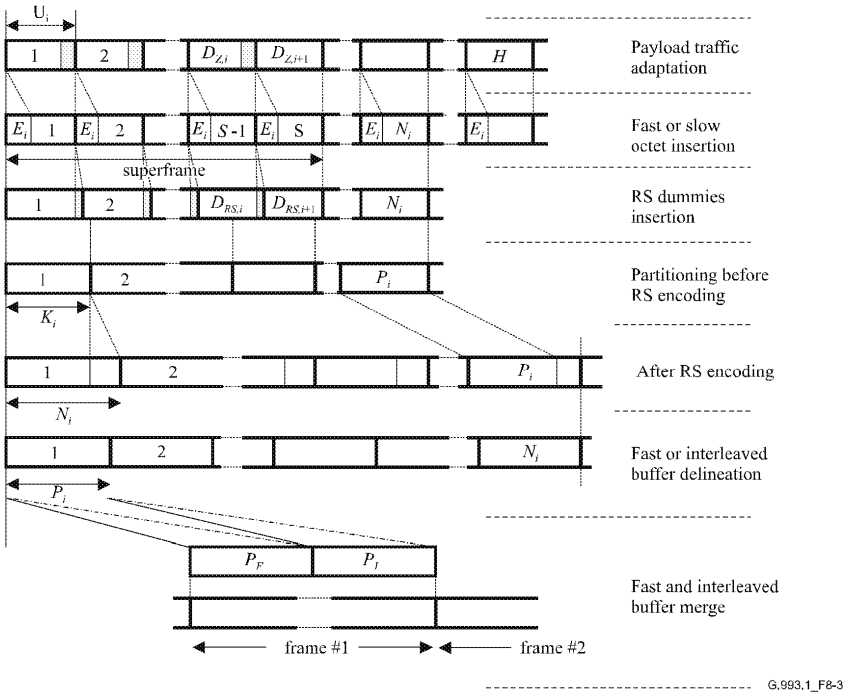

Figure 8-3/G.993.1 – Framing description

8.5.2 Payload adaptation

The α/β interface provides bytes at a rate multiple of 64 kbit/s. In order to map an integer number of bytes into a frame, the TPS-TC byte flow shall be stuffed with the appropriate number of dummies.

For an $n_i \times 64$ kbit/s rate, we have on average $n_i \times 8000/f_s$ bytes per frame, with f_s the symbol frequency. This number will not be integer for a general value of f_s. Since the cyclic extension $L_{CP} + L_{CS} - \beta$ is a multiple of 2^{n+1} however, (see 9.2.2), we always have an integer number of bytes every $H = 138$ frames. If we define k as:

$$k = \frac{8\ kbytes \times H}{f_s}$$

we can transport $n_i \times k$ TPS-TC payload bytes in H frames. In order to transport an integer number of bytes per frame, we have to insert an appropriate number of dummy bytes. Every frame will contain a total of U_i bytes (TPS-TC bytes + dummy bytes), with:

$$U_i = \left\lceil \frac{n_i \times k}{H} \right\rceil$$

The number of dummy bytes $D_{Z,i}$ to be inserted every H packets shall therefore be:

$$D_{Z,i} = \left\lceil \frac{n_i \times k}{H} \right\rceil \times H - (n_i \times k)$$

These dummy bytes shall be inserted in the last position of the first $D_{Z,i}$ packets of U_i bytes in a sequence of H packets. The value of the $D_{Z,i}$ dummies shall be 0x3A.

8.5.3 RS encoding

After payload adaptation, E_i overhead bytes (see 8.5.5) shall be added to the head-end of each packet of U_i bytes (see Figure 8-3). These bytes are called fast and slow bytes for the fast and slow channel respectively. Next, a sequence of N_i packets of $(E_i + U_i)$ bytes shall be RS-encoded. In order to achieve an integer number of RS-codewords per N_i packets, RS-dummy bytes may have to be inserted. The RS-codeword length is equal to the parameter N_i.

The number of RS-encoded bytes, B_i, per N_i packets is given by:

$$B_i = \left[N_i \times (E_i + U_i) + D_{RS,i}\right] \times \frac{N_i}{K_i}$$

In the above equation, the parameter N_i denotes both the number of packets of $(E_i + U_i)$ bytes and also the length of a RS-codeword (in bytes). The parameter K_i is the number of information bytes in an RS-codeword.

The number of RS dummy bytes, $D_{RS,i}$, inserted to carry an integer number of RS-codewords in every N_i frames is given by

$$D_{RS,i} = \left\lceil \frac{N_i \times (E_i + U_i)}{K_i} \right\rceil \times K_i - N_i \times (E_i + U_i)$$

Each one of the $D_{RS,i}$ dummies shall be inserted at the tail-end of the first $D_{RS,i}$ packets of $(E_i + U_i)$ bytes in a sequence of N_i packets (see Figure 8-3). The value of the $D_{RS,i}$ bytes shall be 0xD3.

After RS-dummy insertion, the number of RS-encoded bytes per frame carried in either the fast or interleaved buffer is given by:

$$P_i = \frac{B_i}{N_i} = \frac{N_i \times (E_i + U_i) + D_{RS,i}}{K_i} = \left\lceil \frac{N_i \times (E_i + U_i)}{K_i} \right\rceil$$

NOTE – The parameter $B_i = P_i\ N_i$ represents both the number of bytes in N_i frames (with P_i bytes per frame) and also the number of bytes in P_i codewords (with N_i bytes per codeword). See Figure 8-3.

8.5.4 Definition of superframe

A superframe shall be composed of 10 packets of $U_i + E_i$ bytes.

8.5.5 Contents of fast and slow bytes

Each of the packets in a superframe shall transport E_i overhead bytes, called fast or slow bytes, depending on the channel. The content of these bytes is summarized in Table 8-3. If the fast buffer is empty, the F-EOC bytes shall be transported in the S-EOC bytes. Otherwise, the S-EOC bytes shall be replaced with payload bytes.

There shall be V VOC bytes per packet. They shall always be transported in the slow channel. A setting of $V = 1$ shall be supported, other values for V should be allowed as optional. The value of V shall be exchanged during initialization (see 12.4.6.2.1.1).

If the fast path is active, the NTR byte in the slow channel shall be replaced with a dummy byte. Similarly for the IB bytes.

The fast and slow dummy bytes shall have the value 0xFF.

Table 8-3/G.993.1 – Contents of fast and slow bytes

Packet	Fast bytes		Slow bytes		
	First byte	Other bytes (if any)	First byte	2nd up to (V + 1)st byte	Other bytes (if any)
1	F-CRC	F-EOC	S-CRC	VOC	S-EOC/payload
2	Synch byte	F-EOC	Synch byte	VOC	S-EOC/payload
3-5	IB	F-EOC	IB/dummy	VOC	S-EOC/payload
6	NTR	F-EOC	NTR/dummy	VOC	S-EOC/payload
7-10	Dummy	F-EOC	Dummy	VOC	S-EOC/payload

8.5.5.1 Cyclic Redundancy Check (CRC)

Two cyclic redundancy checks (CRC) – one for the fast buffer and one for the interleaved buffer – shall be generated for each superframe and shall be transmitted in the first packet of the following superframe (see Table 8-3). The CRC byte for the first superframe shall be set to zero.

Eight bits per buffer type (fast or interleaved) and per superframe shall be allocated to the CRC check bits. These bits shall be computed from the k message bits using the equation:

$$crc(D) = M(D)\ D^8 \text{ modulo } G(D)$$

where:

$M(D) = m_0D^{k-1} + m_1D^{k-2} + \ldots + m_{k-2}D + m_{k-1}$ is the message polynomial

$G(D) = D^8 + D^4 + D^3 + D^2 + 1$ is the generating polynomial

$crc(D) = c_0D^7 + c_1D^6 + \ldots + c_6D + c_7$ is the check polynomial

D is the delay *operator*.

That is, crc(D) shall be the remainder when $M(D)D^8$ is divided by $G(D)$.

The bits covered by the crc shall include:

– fast buffer: all bits of the fast buffer before RS encoding, except the crc;

– interleaved buffer: all bits of the interleaved buffer before RS encoding, except the crc.

Each byte shall be clocked into the CRC least significant bit first.

8.5.5.2 Synchronization byte

The synchronization byte has the value 0x3C. This synchronization byte shall be used to monitor the frame synchronization.

8.5.5.3 Indicator Bits (IB)

The indicator bits are used to transmit far-end defects and anomalies. The description of the content of the three indicator bytes shall be as summarized in Table 8-4. If the fast channel is active, the indicator bytes shall be transmitted in this channel and the indicator bytes in the slow channel shall be replaced by dummies (having value 0xFF, see 8.5.5).

Table 8-4/G.993.1 – Content of indicator bits

Byte #	Bit #	Definition
1	b0-b7	Reserved for future use
2	b0	Febe-s
	b1	Ffec-s
	b2	Febe-f
	b3	Ffec-f
	b4	Flos
	b5	Rdi
	b6	Fpo
	b7	Flpr
3	b0	LoM (Loss of Margin)
	b1	Fhec-s (used for ATM only, shall be set to 0 for PTM)
	b2	Fhec-f (used for ATM only, shall be set to 0 for PTM)
	b3	Fncd-s/Focd-s (used for ATM only, shall be set to 0 for PTM)
	b4	Fncd-f/Focd-f (used for ATM only, shall be set to 0 for PTM)
	b5-b7	Reserved for future use

The active state of a bit shall be high (value 1). Bits that are reserved for future use shall be set to low (value 0).

The definition of the anomalies and defects linked to each of the indicator bits can be found in 10.5.4. The LoM-bit shall signal a loss of margin at the far end. It shall become high once loss of margin is detected and shall remain high as long as this condition exists.

8.5.5.4 Network Timing Reference (NTR)

Isochronous services require the same timing reference at transmit and receive sides in higher layers of the protocol stack. To support the transmission of this timing signal, the VDSL system shall transport an 8-kHz timing marker.

For applications that require NTR, NTR shall be transported in the following way:

The VTU-O shall derive a local 8-kHz timing reference (LTR), by dividing its sample clock with the appropriate number. For a VDSL system using $N_{sc} = 2^{n+8}$ tones, the sampling frequency could for instance be $2\ N_{sc}\ \Delta f$ and the dividing factor would then be $69 \times 2^{n+2}$.

The VTU-O shall estimate the change in phase offset between the NTR and the LTR from the previous superframe to the present. This value shall be expressed in cycles of a clock running at

frequency 2 N_{sc} Δf and shall be transported in the NTR overhead byte (see Table 8-3) as a 2's-complement number.

A positive value of the change in phase offset shall indicate that the LTR has a higher frequency than the NTR. A negative value of the change in phase offset shall indicate that the LTR has a lower frequency than the NTR.

The LTR, being proportional to Δf, has a maximum frequency variation of 50 ppm (see 9.2.1.1). The NTR has a maximum variation of 32 ppm. The maximum difference is therefore 82 ppm. This would result in a maximal phase offset of 0.205 µs per superframe. This corresponds to about $0.45.2^n$ samples. For the largest value of n (n = 4), this corresponds to somewhat more than 7 samples (in the positive or negative direction). One byte of information should therefore be sufficient.

8.5.6 Convergence of fast and interleaved buffer

Data from the interleaved and (optional) fast buffer shall be combined so that in each frame there shall first be a segment of fast data followed by a segment of interleaved data. Figure 8-4 illustrates this process.

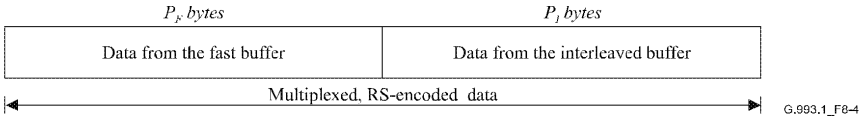

Figure 8-4/G.993.1 – Convergence of the fast and interleaved data into one frame

The total number of RS-encoded bytes per frame, P_{total}, is given by:

$$P_{total} = P_I + P_F$$

where P_I and P_F are the number of RS-encoded bytes from the interleaved and fast paths.

9 PMD sublayer

9.1 PMD functional model

The functional model of the PMD sublayer is presented in Figure 9-1.